Content-gnostic Bitrate Ladder Prediction for Adaptive Video Streaming

Angeliki V. Katsenou*, Joel Sole† and David R. Bull*
*Department of Electrical and Electronic Engineering, University of Bristol, Bristol BS8 1UB, UK
{angeliki.katsenou, dave.bull}@bristol.ac.uk
†Netflix Inc., Los Gatos, California, USA
{jsole}@netflix.com

***Abstract*—A challenge that many video providers face is the heterogeneity of networks and display devices for streaming, as well as dealing with a wide variety of content with different encoding performance. In the past, a fixed bit rate ladder solution based on a "fitting all" approach has been employed. However, such a content-tailored solution is highly demanding; the computational and financial cost of constructing the convex hull per video by encoding at all resolutions and quantization levels is huge. In this paper, we propose a content-gnostic approach that exploits machine learning to predict the bit rate ranges for different resolutions. This has the advantage of significantly reducing the number of encodes required. The first results, based on over 100 HEVC-encoded sequences demonstrate the potential, showing an average Bjøntegaard Delta Rate (BDRate) loss of 0.51% and an average BDPSNR loss of 0.01 dB compared to the ground truth, while significantly reducing the number of pre-encodes required when compared to two other methods (by 81%-94%).**

***Index Terms*—Rate-Quality Convex Hull, Bitrate Ladder, Per-title Video Encoding, HEVC, Adaptive Video Streaming.**

I. Introduction

Most video providers address the increased demands associated with internet video transmission using adaptive streaming. In adaptive streaming, several encoded versions of each video sequence, at different resolutions and bit rates, are processed with the aim of satisfying the requirements and limitations of each end user depending on the network bandwidth and display specifications. For example, a different version of the same video sequence will be streamed to a client's mobile device (e.g. 1920×1080 at 30 fps) compared to that on his/her smart TV (e.g. 3840×2160 at 60 fps). From a provider's perspective, the aim is to deliver the best video quality at the lowest possible bit rate while maintaining a satisfying viewing experience.

To address this challenge, the traditional approach suggests to build a fixed "bit rate ladder" that would indicate the recommended spatial resolution for the available bit rate. The traditional bit rate ladder is either content-agnostic, with fixed bit rate ranges allocated per resolution, e.g. [1], or employs limited classification of the content based on genre, e.g. [2]. Recent research [3]–[11] has focused on moving beyond the fixed "bitrate ladder" approach traditionally followed in fixed rate encoding systems. For example, the idea behind the approach used by Netflix [3], [4] is to obtain the Rate-Quality (RQ) curves per title at different resolutions and different bitrates by running several trial encodings at different quantization levels. Then, this information is used to construct the convex hull of the RQ curves (particularly scaled PSNR-rate in [3] and scaled VMAF-rate in [4]) and hence to reveal the optimal parameters for the available bit rate range.

A similar approach is presented in [5], where the authors use measurements on the actual usage of millions of video clips to create probability distributions of available bandwidth and viewport sizes. These probability distributions feed the optimization process that ensures video quality preservation while reducing the required bit rate compared with existing techniques. Other per-title-encoding approaches have been offered in the industry: from Bitmovin [8], MUX [9], CAMBRIA [10] and more [11]. All of these approaches are compared in [11]. The Bitmovin solution [8] and CAMBRIA [10] calculate the encoding complexity. According to the first [8], a complexity analysis is performed on each incoming video. This analysis results in a variety of measurements being processed by machine-trained model to adjust the encoding profile to match the content. The CAMBRIA solution estimates the encoding complexity by running a fast constant rate factor encoding [10]. MUX [9] introduced a deep-learning based approach that takes as input the vectorized video frames and predicts the bit rate ladder. While all the above solutions are interesting, they are proprietry and hence it is not possible to provide further details nor to make direct comparisons.

Most of the aforementioned solutions rely on massive numbers of encodes; hence the computational, energy and financial costs are high, since cloud encoding services are usually employed [12]. In this paper, we propose a content-gnostic method that predicts the cross-over points associated with the individual RQ curves and provides a content-customized estimation of the bit rate ladder. The method extracts spatio-temporal features and statistics from sequences at their native resolution and then, by employing machine learning methods, it predicts the quantization levels (or quantization parameters (QPs)) at which the RQ curves across the different resolutions intersect. Based on this prediction, only a small number of encodes needs to be performed in order to determine the bit

The work presented was supported by the Leverhulme Early Career Fellowship (ECF-2017-413) and by Netflix Inc. We also gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

Fig. 1: Sample frames of the considered dataset [18].

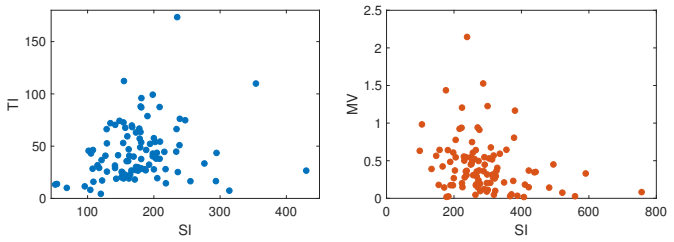


Fig. 2: Scatter plots of the SI-TI and SI-MV features of the considered 4K dataset.

rates at which resolutions should be switched, significantly reducing the extensive number of encodings required by state-of-the-art methods at different resolutions and QPs. This is particularly helpful for adaptive streaming applications as an estimate of the bit rate ladder per-title can be provided at a much lower cost.

The paper has the following structure. In Section II the dataset is described along with its content characteristics. The proposed methodology is detailed in Section III as well as the evaluation results. Finally, conclusions and future work are outlined in Section IV.

II. Description of the Dataset

It is important for the present work to have a big video data set that covers a variety of scenes. Therefore, we used a dataset of 100 publicly available UFHD video sequences from different sources: Netflix Chimera [13], Ultra Video Group [14], Harmonic Inc [15], SJTU [16] and AWS Elemental [17]. The same dataset was also used as a training dataset in [18], [19]. Example frames of the dataset are depicted in Fig. 1.

All the sequences were spatially cropped to 3840×2160 (if originally, the resolution was 4096×2160 pixels), converted to $4:2:0$ chroma subsampling (if originally otherwise). Also, two of the sequences were temporally downsampled from 120 to 60fps in order to match the majority frame rate of 60 fps. Finally, the sequences were temporally cropped to 64 frames. Each sequence contains a single scene (without scene cuts) and the majority of the test sequences have a frame rate of 60 fps and a bit depth of 10 bits per sample.

To illustrate the variety of video content found in our training dataset, first, we extracted the spatial (SI) and temporal (TI) information, as well as the mean magnitude of the motion vectors (MV) at 4K resolution, as recommended by Winkler in [20]. As can be seen in the scatter plots in Fig. 2, the selected sequences features show a wide coverage of the spatio-temporal domain.

III. Content-driven Prediction of the Cross-over Points

In this work, our goal is to predict a content-gnostic bitrate ladder per input video. To this end, we first extract spatio-temporal features extracted from uncompressed video sequences only at the native resolution. Then, compressing the test sequences for a range of QPs across different resolutions, we construct the ground truth by constructing the convex hull and computing the cross-over points. Using the extracted spatio-temporal features and the real cross-over points, we train machine learning models to perform regression and predict the cross-over QPs of the different resolution RQs. This means that no pre-encodings are required with the proposed method to estimate the cross-over QPs. The only pre-encodings required take place after the prediction of the cross-over QPs to fully define the bitrate and quality ranges per resolution. A block diagram that briefly outlines the proposed method is illustrated in Fig. 3.

A. Constructing the Convex Hull and defining the Cross-over QPs as Ground Truth

In order to perform machine-learning based regression, it is important first to construct the ground truth, aka the real convex hulls, and define the cross-over QPS for the different resolutions of the considered dataset. We considered the dataset described in Section II and spatially downscaled all sequences (see Fig. 3) using Lanczos-3 filter [21], as implemented by FFmpeg [22], at two different resolutions, from 4K - 2160p (native) to FHD - 1080p and HD - 720p. Then, we encoded all the different versions of the sequences with the HEVC reference software HM16.20 [23] with a Random Access profile, a 64 frames long Intra period, a group of picture length of 16 frames, and using the following QP ranges: $QP_{4K} = \{21, 22, \ldots, 45\}$, $QP_{FHD} = \{17, 18, \ldots, 39\}$, and $QP_{HD} = \{15, 16, \ldots, 35\}$. The reason behind the slightly shifted QP ranges is that we wanted to ensure that the RQ curves would intersect. As can be seen in Fig. 3, after decoding the sequences, we upscaled them in the 4K resolution using the same filter, as this is intended to be the displayed resolution (see Fig. 3). All quality metrics are computed at the display resolution, aka 4K, as recommended also in [24].

In Fig. 4 (a), we have plotted the RQ curves, in terms of PSNR versus a logarithmic scale of the bitrate, for all the considered sequences at three spatial resolutions, at native 4K, at 4K upscaled from FHD and at 4K upscaled from HD[1], and for the same range of quantization levels, i.e. $QP = \{21:2:35\}$. We have also plotted the convex hulls of the RQ curves to better visualise the wide range of the selected sequences.

[1]Henceforth, the FHD and HD resolutions will refer to the sequences that have been encoded at HD and SD, respectively, and then were upscaled to 4K.

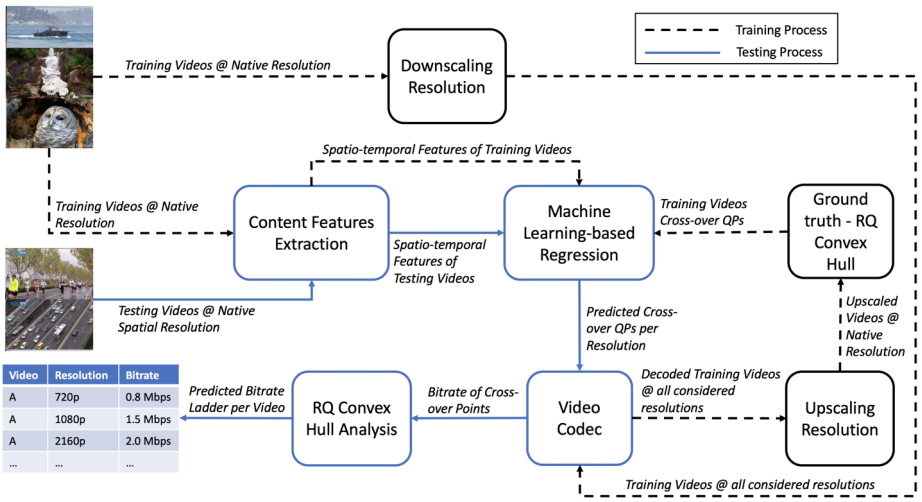
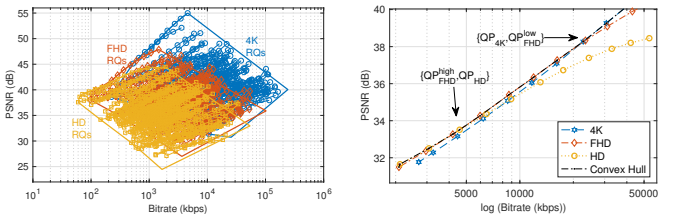
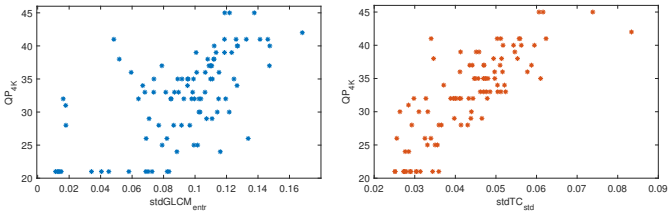


Fig. 3: Diagrammatic overview of the proposed method.



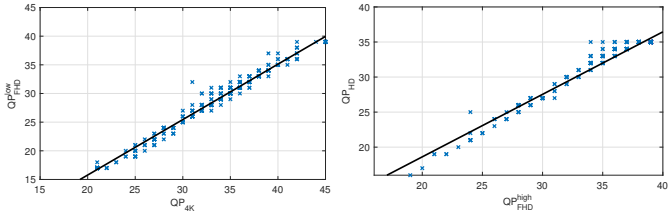
(a) RQ curves of the considered dataset. (b) Theoretical intersections of RQ curves.

Fig. 4: Overview of the considered dataset RQ curves for the same QP range at three different resolutions and an example of intersecting curves.



(a) QP_{4K} vs stdGLCM$_{\text{ent}}$. (b) QP_{4K} vs stdTC$_{\text{std}}$.

Fig. 5: Example of content dependency of the cross-over QPs.



(a) QP^{low}_{FHD} against QP_{4K}. (b) QP_{HD} against QP^{high}_{FHD}.

Fig. 6: Scatter plots of cross-over QP pairs.

The first observation is that the wide range of content features is verified by the high diversity in RQ curves. Furthermore, it can be easily seen that there is a shift of the RQ curves range to lower quality and bitrate in respect with the downscaled spatial resolution. The intersection points differ per sequence and highly depend on the content characteristics.

A typical example of intersecting RQ curves is drawn in Fig. 4 (b). For the theoretical estimation of the cross-over points of the RQ curves using the developed models, we just need to find the QPs at which the curves intersect. We define the intersection points as pairs of cross-over QPs, (QP_{resA}, QP_{resB}), where A and B are the intersecting curves of resolution A and B. For our use case, as explained earlier, we consider three resolutions 4K, HD, and SD. Thus, we consider two pairs of QPs, namely $(QP^{low}_{FHD}, QP_{4K})$ and $(QP_{HD}, QP^{high}_{FHD})$. We have to note, however, that there are cases the expected resolutions do not intersect. For example, for a few sequences, the HD RQ curve is always on top of the 4K RQ and for other cases the HD RQ does not intersect with the SD RQ.

B. Relation of Cross-over QPs

As mentioned above, the cross-over QPs are content-dependent. This becomes evident when we plot the ground truth cross-over QPs against the spatio-temporal features extracted from the video sequences at their native resolution. Such examples are illustrated in Fig. 5, where the QP_{4K} is scattered against two of the extracted spatio-temporal features stdGLCM$_{\text{ent}}$ and stdTC$_{\text{std}}$ (see Table I for more details).

In Fig. 6, the relation of the pairs of cross-over QPs are demonstrated. It can be seen that the cross-over points cover a wide range of approximately 25 different values. We have also explored the relation of the cross-over QPs and found that they are linearly correlated with a Pearson Linear Correlation Coefficient (LCC) value equal to 0.9917 and a Spearman Rank Correlation Coefficient (SROCC) 0.9888 for the $(QP^{low}_{FHD}, QP_{4K})$ pair. For the $(QP_{HD}, QP^{high}_{FHD})$ pair, LCC is equal to 0.9817 and SROCC 0.9538. The slightly lower correlation values in the second pair are a result of the clipping within the considered range of values. The linear relation of the cross-over QPs, as well as the clipping effect in $(QP_{HD}, QP^{high}_{FHD})$ pair can also be visually verified by the scatter plots in Fig. 6.

C. Predicting the cross-over QPs

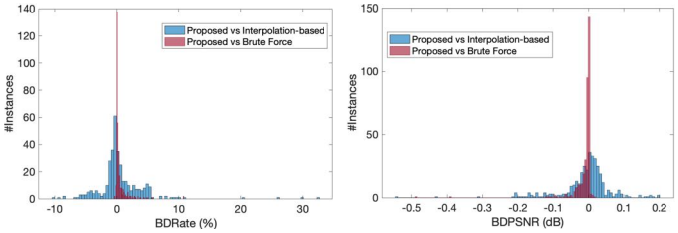
For the validation of the proposed method, we considered two other methods to compare against. First, we used the

TABLE I: List of features and their statistics.

Feature	Statistics
Grey-Level Co-occurrence Matrix (GLCM) [28]	F1.meanGLCM$_{con}$, F6.stdGLCM$_{con}$, F2.meanGLCM$_{cor}$, F7.stdGLCM$_{cor}$, F3.meanGLCM$_{hom}$, F8.stdGLCM$_{hom}$, F4.meanGLCM$_{enr}$, F9.stdGLCM$_{enr}$, F5.meanGLCM$_{ent}$, F10.stdGLCM$_{ent}$
Temporal Coherence (TC) [26]	F11.meanTC$_{mean}$, F16.stdTC$_{mean}$, F12.meanTC$_{std}$, F17.stdTC$_{std}$, F13.meanTC$_{skw}$, F18.stdTC$_{skw}$, F14.meanTC$_{kur}$, F19.stdTC$_{kur}$, F15.meanTC$_{entr}$, F20.stdTC$_{entr}$
Predicted cross-over QPs	F21.$\hat{QP}_{4K}$, F22.$\hat{QP}_{FHD}^{low}$, F23.$\hat{QP}_{FHD}^{high}$

TABLE II: Selected features & validation metrics of predicted cross-over QPs.

QP	Sel. Features	LCC	SROCC	R^2	MAE	RMSE
$\hat{QP}_{4K}$	F2, F4, F5, F11, F12, F14	.9350	.9164	.91	1.41	1.96
$\hat{QP}_{FHD}^{low}$	F2, F4, F5, F11, F12, F14, F21	.9442	.9296	.90	1.35	1.96
$\hat{QP}_{FHD}^{high}$	F2, F4, F5, F11, F12, F14, F22	.9536	.9076	.91	.95	1.36
$\hat{QP}_{HD}$	F2, F4, F5, F11, F12, F14, F21, F22, F23	.9531	.8751	.92	.76	1.15



(a) BDRate histograms. (b) BDPSNR histograms.

Fig. 7: Histograms of the resulting BDRate and BDPSNR of the proposed over the compared methods.

brute force method to construct the convex hull per Group of Pictures (GoP) of the 100 considered sequences, as also defined in Section III-A, and found the cross-over QPs that will be considered as the ground truth. The brute force method theoretically creates the optimal convex hull. Additionally, we considered a more practical method to compare against, by limiting the number of pre-encodes in 7 per resolution (using equidistant QPs to cover the range) and by using a piece-wise cubic Hermite interpolation for the in-between QPs. This method of course results in constructing a suboptimal convex hull, but can provide a good approximation of it, while significantly reducing the number of pre-encodes.

For the encoding of the sequences we used the latest version of HEVC reference software, HM16.20 with Random Access profile according to the Common Testing Conditions [23], [25]. Then, we extracted spatio-temporal features from the uncompressed 4K sequences and started sequentially (with a descended order of the considered spatial resolutions) predicting the cross-over QPs. A subset of the considered features is presented in Table I. The literature is rich with spatio-temporal features to describe content. We decided to consider the specific features, as they have been proved useful in our previous work for video analysis, RQ prediction and more [18], [26], [27].

Prior to each QP prediction, we applied feature selection, and particularly recursive feature elimination, on the set of spatio-temporal features. For the QP_{4K} prediction, we only relied on spatio-temporal features. For the rest of the predictions, we made use of the identified relations and considered the previously predicted QPs (of the highest resolutions). Hence, the predictions were performed sequentially by four different Gaussian Processes (GP) with rational quadratic kernels. We have tested other methods as well, such as Support Vector Machines with different kernels, Random Forests, etc, but GP was the best performing one for this work. To avoid overfitting, we deployed a ten-fold random cross-validation process. The results shown in the next section are the outcome of the ten-fold cross-validation and the accuracy of prediction metrics are averaged over the ten folds.

1) Results: The selected features and accuracy of prediction are given in Table II. Regarding the selected features, we observe that the same spatio-temporal features were selected for all four QPs. Additionally, the previously predicted QPs were selected verifying the identified relations of the cross-over QPs. Regarding the accuracy of prediction of the tested models, as can be seen, it can be considered as high due to the high R^2 values (greater or equal to 0.9). In addition to this, the cross-correlation metrics LCC and SROCC between the predicted and the ground truth QPs are high[2]. Also, the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) are considerably low and comparable for all predicted QPs.

To assess whether the predicted QPs can result in building RQ convex hulls similar to the optimal ones, we computed the Bjøntegaard Delta metrics between the proposed method and two other methods, the brute force that results in the optimal the convex hull and the interpolation-based method that uses significantly fewer pre-encodes. The results are illustrated in Fig. 7. As can be seen, both BD metrics are reasonably low with an average BDRate loss of 0.5093% for the brute force method and an average loss of 0.6249% for the interpolation-based method. The BDPSNR is also low with an average BDPSNR loss of 0.0119 dB loss for the comparison against the brute force method and an average loss of 0.0098dB for the interpolation-based method. The proposed method achieves a closer to the optimal performance as for the majority of the sequences, 85%, it resulted in BDRate losses below 1% (when compared against the brute force method). It is also noticeable, that the distribution of the BD metric values are different for the interpolation-based method, where we notice both losses and gains. This is attributed to the

[2]We have to note that the predicted values were rounded to the nearest integer and clipped to the range of QP values, before computing the correlation metrics.

TABLE III: Comparison of the number of pre-encodings required per method.

Method	# Pre-encodings	Presented Test Case
Brute Force	#resolutions×#QPs	69
Interpolation-based	#resolutions×7	21
Proposed	(#resolutions-1)×2	4

numeric interpolation errors that result in suboptimal convex hulls (over- or underestimated) compared to the brute force method.

2) Discussion on the complexity: The benefit that the proposed method has to offer is that it can significantly reduce the number of pre-encodings required to build a bitrate ladder tailored for each video. In Table III, a comparison of the number of pre-encodings required from the brute force method is reported. The brute force method can produce the optimal convex hull of the RQs across resolutions, thus define the optimal bitrate ladder, as also indicated in [4]. As can be seen the number of pre-encodings required are significantly reduced compared to both methods: 94.2% fewer pre-encodings compared to the brute force method and 80.95% compared to the interpolation-based method.

To provide a better understanding of the overhead that the feature extraction imposed in our test case, we have assessed its computational overhead. The ratio of the average feature extraction time for a sequence at the 4K resolution to the average 4K encoding time for a sequence at one QP is 0.18.

IV. Conclusion

The computational and financial cost of converting the traditional bit-rate ladder into a content-customised solution is enormous. Therefore, we proposed a method that can predict the bitrate ranges of the considered resolutions based on spatio-temporal features extracted from the uncompressed videos at their native resolution and with a few video encodings (two encodes per RQ intersecting points). The first results are promising reaching only to a mean BDRate loss of 0.51% and a mean BDPSNR of 0.01dB compared to the ground truth, while requiring 94.2% and 81% fewer pre-encodes compared to the brute force method and the interpolation-based method, respectively. In the future, our focus will be on further improving the proposed method to predict the cross-resolution RQ curves cross-over points at a floating point accuracy. Furthermore, the method will be extended to efficiently predict the bitrate ladder per shot and title.

References

[1] Apple, "HLS Authoring Specification for Apple Devices," https://developer.apple.com/documentation/http_live_streaming/hls_authoring_specification_for_apple_devices.

[2] S. Lederer, C. Müller, and C. Timmerer, "Dynamic Adaptive Streaming over HTTP Dataset," in *Proceedings of the 3rd Multimedia Systems Conference*, 2012, ACM MMSys, pp. 89–94.

[3] J. De Cock, Z. Li, M. Manohara, and A. Aaron, "Complexity-based consistent-quality encoding in the cloud," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 1484–1488.

[4] I. Katsavounidis, "Dynamic optimizer - a perceptual video encoding optimization framework," https://medium.com/netflix-techblog/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f.

[5] C. Chen, Y. Lin, S. Benting, and A. Kokaram, "Optimized Transcoding for Large Scale Adaptive Streaming Using Playback Statistics," in *25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 3269–3273.

[6] A. Zabrovskiy, C. Feldmann, and C. Timmerer, "A practical evaluation of video codecs for large-scale http adaptive streaming services," in *25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 998–1002.

[7] L. Toni, R. Aparicio-Pardo, K. Pires, G. Simon, A. Blanc, and P. Frossard, "Optimal Selection of Adaptive Streaming Representations," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 11, no. 2s, pp. 43:1–43:26, Feb. 2015.

[8] Bitmovin, "White Paper: Per Title Encoding," https://bitmovin.com/whitepapers/Bitmovin-Per-Title.pdf, 2018.

[9] MUX, "Instant Per-Title Encoding," https://mux.com/blog/instant-per-title-encoding/.

[10] Cambria, "Feature: Source Adaptive Bitrate Ladder (SABL)," https://www.capellasystems.net/capella_wp/wp-content/uploads/2018/01/CambriaFTC_SABL.pdf.

[11] J.Ozer, "Per-Title Encoding Comparison: Crunch Video Optimization Technology compared to: Brightcove CAE, Capped CRF, Capella Systems SABL, JWPlayer," https://streaminglearningcenter.com/wp-content/uploads/2018/07/Report_final.pdf.

[12] J. Ozer, "A Cloud Encoding Pricing Comparison," http://docs.hybrik.com/repo/cloud_pricing_comparison.pdf.

[13] I. Katsavounidis, ""chimera" video sequence details and scenes," https://www.cdvl.org/documents/NETFLIX_Chimera_4096x2160_Download_Instructions.pdf, 2015.

[14] T. U. Ultra Video Group, ," http://ultravideo.cs.tut.fi/, [Online; accessed 2017-01-23].

[15] Harmonic Inc 4K demo footage, ," https://www.harmonicinc.com/4k-demo-footage-download/, [Online; accessed 2017-05-01].

[16] L. Song, X. Tang, W. Zhang, X. Yang, and P. Xia, "The sjtu 4k video sequence dataset," in *Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*. IEEE, 2013, pp. 34–35.

[17] A. Elemental, ," https://www.youtube.com/playlist?list=PLwIpNYl7S0G_C5I76Tf46n6ImKssMn2kT/, [Online; accessed 2018-07-02].

[18] M. Afonso, F. Zhang, and D. R. Bull, "Spatial resolution adaptation framework for video compression," in *SPIE Optical Engineering + Applications*, 2018, vol. Proceedings Volume 10752, Applications of Digital Image Processing XLI.

[19] M. Afonso, F. Zhang, A. Katsenou, D. Agrafiotis, and D. Bull, "Low complexity video coding based on spatial resolution adaptation," in *IEEE International Conference on Image Processing (ICIP)*, 2017.

[20] S. Winkler, "Analysis of public image and video databases for quality assessment," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 6, pp. 616–625, 2012.

[21] C. E. Duchon, "Lanczos filtering in one and two dimensions," *Journal of Applied Meteorology*, vol. 18, no. 8, pp. 1016–1022, 1979.

[22] "FFMPEG," https://www.ffmpeg.org.

[23] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[24] J. Sole, L. Guo, A. Norkin, M. Afonso, K. Swanson, and A. Aaron, "Performance comparison of video coding standards: an adaptive streaming perspective," https://medium.com/netflix-techblog/performance-comparison-of-video-coding-standards-an-adaptive-streaming-perspective-d45d0183ca95, 2018.

[25] K. Sharman and K. Sühring, "Common Test Conditions for HM video coding experiments," Tech. Rep., document JCTVC-AC1100 of JCT-VC, Oct. 2017.

[26] A. Katsenou, M. Afonso, D. Agrafiotis, and D. R. Bull, "Predicting Video Rate-Distortion Curves using Textural Features," in *Picture Coding Symposium (PCS)*, Dec 2016.

[27] A. V. Katsenou, T. Ntasios, M. Afonso, D. Agrafiotis, and D. R. Bull, "Understanding video texture - a basis for video compression," in *Multimedia Signal Processing (MMSP), IEEE 19th International Workshop on*. IEEE, 2017, pp. 1–6.

[28] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, Nov 1973.